Extending Mobile Devices by Exploiting Remote Resources

Timo Hönig (PhD student), Rüdiger Kapitza and Wolfgang Schröder-Preikschat
Department of Computer Science, University of Erlangen-Nuremberg, Germany
{hoenig,kapitza,wosch}@cs.fau.de

I. Introduction and Motivation

Mobile devices such as laptops, handhelds and mobile phones share a common limitation: battery life. Even so today's operating systems have a sophisticated power management stack enabling to exploit the battery energy in the most efficient manner, battery life of mobile devices has not kept pace with technological progress in other fields – such as clock speed, storage and network bandwidth (Table I).

Whereas other efforts focus on improving power management in the context of isolated systems [1] we target to address this increasing gap by extending the execution scope of mobile devices to the permanent infrastructure. Thereby, we intent to exploit remote resources of the distributed network and expand on the early ideas of cyber foraging [2].

Today, network connectivity is a key factor for mobile devices. The characteristics of network links providing this connectivity have undergone a drastic change over the last couple of years.

Permanence. Previous network links were temporary and activated on demand in order to limit connectivity costs to the essential minimum. Today's mobile devices usually have at least one permanent link – for example, WiFi, GSM or 3G. This change invited application developers to implement a wide range of different features which rely on a more or less permanent connection.

Bandwidth. In the last 30 years, the bandwidth of network connections has increased by the factor of 10^6 (Table I). This change sped up the transition from local I/O to network I/O. At acceptable latency, network I/O at present provides the basis for data safety (e. g., continuous backup services [3]) and flexible data accessibility (e. g., multimedia streaming).

Expenses. Time and traffic based billing models have more or less been replaced by cheap flat-fee models. Accordingly, the use of mobile devices has changed substantially towards the extensive use of the Internet.

Power consumption. In contrast to the outlined characteristics above, one characteristic has changed to the worse: power consumption. The utilization of a wireless network link is causing high power consumption. Additionally, the active state of a network link causes other components like processing units, memory controllers and storage modules to drain energy of the system's battery. In total the power consumption of those devices can exceed the power consumption of the wireless link itself.

Confronted with the outlined changes, operating system designers are facing new challenges regarding more efficient and new ways of power management. This is especially the case as the promised revolution in battery technology has not taken place so far and there is limited reason to believe this will change in the near future. As today's usage profiles of mobile devices demand for permanent network connections we cannot ignore these challenges and accordingly identified ways to address them.

Year	Clock Speed	RAM	Storage	Bandwidth	Battery
1981	4 MHz	4 Kbyte	360 Kbyte	300 bit/s	1 h
1991	16 MHz	2 Mbyte	80 Mbyte	56 Kbit/s	2 h
2001	750 MHz	256 Mbyte	20 Gbyte	100 Mbit/s	4 h
2010	2 x 2.4 GHz	1 Gbyte	160 Gbyte	1 Gbit/s	10 h

TABLE I: Technology trends of mobile devices

II. Early Bird Approach

Our concept introduces a new component within a distributed network. Each node of the network is either limited by energy constraints (e. g., mobile devices) or not. Limited nodes need to exploit their available energy in the most efficient way in order to offer a convenient user experience. This endeavor can be supported by nodes which are not limited by energy constraints. Such a node, the so-called *Early Bird*, is the pivot of our concept.

As the Early Bird does not suffer from power constraints it can undertake tasks normally performed by the limited mobile devices. To do so the Early Bird acts as a supervisor for the participating mobile devices. In order to operate as a supervisor the Early Bird partially tracks the state of the mobile devices and closely monitors their interaction.

When a network node is issuing a request towards a monitored mobile device the associated Early Bird intercepts and analysis the request. In case the Early Bird possesses enough information about the current runtime state of the addressed device it can process the request without passing it to the targeted node. As a result of that the mobile device can remain in a sleep state and save valuable energy.

To retain network transparency for applications and closely coordinate local with distributed energy management we target the implementation of the Early Bird concept as an extension to the operating system. At this level we can also efficiently respond to connectivity changes.

III. Energy Savings

Utilizing the Early Bird concept helps to save a reasonable amount of energy which extends the battery life of participating mobile devices. Communication within a peer-to-peer network is well-suited to support our claims as it requires that participants not only receive but also send data. This causes much higher energy usage compared to an asynchronous communication model using a centralistic network infrastructure (e. g., fetching emails).

By introducing the Early Bird as a supervisor for the mobile devices participating in the peer-to-peer network we can minimize the impact on their battery life. The Early Bird is tracking the data flow between the network nodes and caches all payload data. Whenever a participant of the peer-to-peer network requests a chunk of data from a mobile device, the Early Bird can block this request. It subsequently handles the requests on behalf of the mobile device.

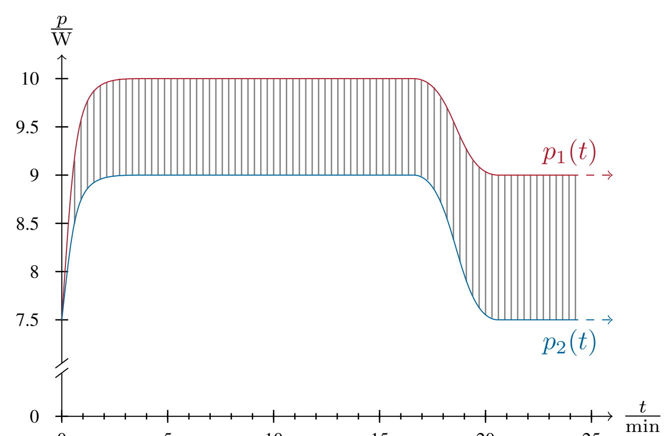

FIGURE I: Anticipated savings for peer-to-peer environments

By this operation the network characteristics change for the mobile device. As the Early Bird will be able to answer requests directed at the mobile device, most of the time the mobile device is not required to utilize the network link for sending data. The mobile device will only be required to utilize the network link for receiving data, exactly as in a centralistic network infrastructure.

When the download operation for the mobile device has come to an end, the Early Bird will continue the participation in the peer-to-peer environment on behalf of the mobile device. This ensures that the operation of the peer-to-peer network itself will not be affected in a negative way.

We have calculated the anticipated energy savings for a common netbook. The system consumes $P_{idle} = 7.5$W when idle, $P_{rx} = P_{tx}$ = 9W when either receiving or sending data and $P_{rxtx} = 10$W when receiving and sending data.

The specified operation is to receive the amount of 100 Mbyte of data at an average download rate of 100 Kbyte/s. Therefore the task is accomplished after 16.6 minutes. The two graphs (Figure I) visualize the power consumption during this time. The first graph $p_1(t)$ shows the energy consumption of the netbook without the Early Bird, while the second graph $p_2(t)$ shows the energy consumption with the Early Bird operating. The expected energy savings are calculated by subtracting $p_2(t)$ from $p_1(t)$ over the time.

For participating in the peer-to-peer network, the netbook will *not* have to go into state P_{rxtx} only if the network environment is being supported by an Early Bird. This is because the Early Bird has previously cached all data and handles all requests towards the netbook autonomously. In concrete numbers we expect an energy reduction for the above task of 29.2 Wmin which is equivalent to an increase of the system's battery life of no less than 12%.

IV. Status and Future Work

This poster abstract gives an example how the Early Bird concept helps resource-limited mobile devices to save energy when participating in a peer-to-peer network. This basic approach can be extended for more complex scenarios. For example, energy usage in mesh networks could be improved by applying the Early Bird mechanism.

Currently we are looking into possibilities where the Early Bird concept is being used for remote execution, where resource-intensive tasks (network I/O- and CPU-bound) are being performed on the Early Bird rather than on the resource-limited mobile device. As the Early Bird is partially tracking the mobile devices' state, this information can be used for purposes such as high availability and data safety. This stands in contrast to approaches where a complete replica is being used [3]. Additionally, the Early Bird will be utilized for speculative execution of pending tasks of the mobile device. The most efficient code path could be pre-calculated and used as input for scheduling decisions on the mobile devices.

References

[1] D. C. Snowdon, E. L. Sueur, S. M. Petters, and G. Heiser, "Koala: A platform for OS-level power management," in *4th EuroSys Conference*, 2009.

[2] R. Balan, J. Flinn, M. Satyanarayanan, S. Sinnamohideen, and H.-I. Yang, "The case for cyber foraging," in *10th ACM SIGOPS European Workshop*, 2002.

[3] B.-G. Chun and P. Maniatis, "Augmented smartphone applications through clone cloud execution," in *12th Workshop on Hot Topics in Operating Systems*, 2009.